

iv enanparq

Encontro da Associação Nacional de Pesquisa e Pós-Graduação em Arquitetura e Urbanismo
Porto Alegre, 25 a 29 de Julho de 2016

SISTEMA HIDRA(!): RECOMBINANDO INTERFACES

SESSÃO TEMÁTICA: PROJETO DIGITAL E FABRICAÇÃO NA ARQUITETURA:
ENSINO, PESQUISA E DESAFIOS

Sandro Canavezzi de Abreu
Universidade Federal de Minas Gerais - UFMG
sandroid@ufmg.br

SISTEMA HIDRA(!): RECOMBINANDO INTERFACES

RESUMO

Nesse artigo trataremos do Sistema Hidra(!), um sistema que potencializa a combinação entre elementos que compõem a internet das coisas, computação física e Arquitetura, ampliando as possibilidades na criação de espaços interativos. Inicialmente, para entendermos o Sistema Hidra(!) como máquina recombinante e parametrizante, que propicia a geração de híbridos entre o digital e o analógico, abordaremos o caráter potencial (possibilidade latente e pré-determinada) da combinatória no contexto dos meios digitais, mais especificamente em processos de digitalização, mapeamento e programação. A compreensão da dimensão combinatorial dos meios digitais, inserida no contexto do que chamamos de “coerência computacional” é condição fundamental para o entendimento do funcionamento e do papel hibridizante do Sistema Hidra(!). Nesse processo de construção do conceito de combinatória e seu contexto, um conceito estruturante será problematizado: a informação. A natureza “descorporificada” da informação propicia que ela assuma qualquer corpo. Essa qualidade é fundamental para a possibilidade de mapeamentos e parametrizações, processos esses que conformam a essência do Sistema Hidra(!). Em seguida entraremos em detalhes dos aspectos técnicos desse sistema, identificando-as como sendo expressões do conceito de combinatória. Finalmente, descreveremos experimentos que ampliam o Sistema Hidra(!), tensionando-o na direção de eventos tangíveis e interativos.

Palavras-chave: Interfaces recombinantes. Combinatória. Parametrização.

HIDRA(!) SYSTEM: RECOMBINING INTERFACES

ABSTRACT

In this article we will introduce the Hidra(!) System, a System that leverages the combination of elements that make up the internet of things, physical computing and architecture , expanding the possibilities in the designing of interactive spaces. Initially, in order to understand the Hydra System (!) as recombinant and parameterizing machine, (which facilitates the generation of hybrid between digital and analog), we will discuss the potential aspect (as latent and predetermined possibility) of combinatorics in the context of digital media, more specifically relating scanning processes, mapping and programming. The understanding of the combinatorial dimension of digital media, inserted in the context of what we call "computer coherence" is a prerequisite for understanding the functioning and the hybridizing role of Hydra System(!). In this process of building the concept of combinatorics and its context, a structural concept will be problematized: information. The "disembodied" nature of information provides it the possibility of taking any body. This ability is essential for the possibility of mappings and parameterizations, which are part of the essence of the Hydra System (!). Then we go into details of the technical aspects of the system, identifying them as expressions of the concept of combinatorics. Finally, we describe experiments that extend the Hydra System (!), tensing it toward tangible and interactive events.

Palavras-chave: Recombining interfaces. Combinatorics. Parametrization.

1. INTRODUÇÃO.

Com a ampliação das possibilidades da utilização de sensores e atuadores em Arquitetura, compondo edificações e ambientes que dialogam com a “internet das coisas” e computação física¹, faz-se necessário a construção de estratégias que potencialize a combinação e recombinação entre elementos arquitetônicos e sensores/atuadores. A combinação entre esses elementos pode ter motivações que buscam maior sustentabilidade da edificação, maior eficiência energética e maior conforto do usuário.

Nesse contexto, O Sistema Hidra se propõe como uma máquina recombinante e parametrizante, que gera híbridos entre o digital e o analógico, potencializando a geração de híbridos entre elementos arquitetônicos e elementos da computação física. Para entendermos os Sistema Hidra(!) como tal, precisamos nos aprofundar nos processos de digitalização e processamento da informação.

Independentemente da velocidade de processamento de um sistema digital, do fato de o sistema ser binário ou quântico, ou do nível de complexidade dos cálculos e algoritmos, sempre haverá a relação de um sistema discreto com um contínuo/físico (analógico). E essa relação sempre se dará tendo como base a combinatória. Assim posto, da digitalização interessa o que poderíamos chamar de “dimensão arbitrária” e as formas da sua “aparição”, isto é, o mecanismo que efetiva a conversão de entidades analógicas (físicas) em entidades digitais (numéricas/binárias/elétricas) e vice-versa: a combinatória.

Em um primeiro momento, iremos mostrar como a combinatória é utilizada como artifício tradutor, atentando-se para sua ambivalência: ela é **reduzora** e, ao mesmo tempo, um **motor hibridizante** que, em último caso, poderia ser considerado como aliado de um processo criativo. Ela é redutora, pois para a sua existência e efetivação, ela faz parte de um conjunto de processos que traduzem, como já dito, o universo analógico/físico em números. Esse processo transforma entidades contínuas (ondas, por exemplo) em “partículas” numéricas (números binários). Essa atividade discretiza e fragmenta coerentemente o universo físico. Ressalta-se aqui a ideia de coerência: a fragmentação tem que ser organizada de forma a ser assimilada pelo computador. A essa coerência chamamos de “coerência computacional” e dela trataremos mais adiante.

¹ Computação física: aquela voltada para a obtenção de informações de um ambiente e seus usuários a partir da utilização de sensores. Visa também estabelecer meios físicos para controlar ações nesse ambiente a partir de atuadores. A internet das coisas alia computação física com a possibilidade de conectar esses elementos à internet.

A tradução “em números” não vem sem um custo, pois sempre se perde informação ao se digitalizar algo analógico, uma vez que o mundo físico é convertido em números a partir de taxas de amostragem: recortes no tempo e no espaço que convertem o infinito entre dois pontos em quantias mensuráveis. Intensidades que variam continuamente são fragmentadas em degraus abruptos. A quantidade de recortes que se promove no objeto analógico é o que chama-se de resolução de um sistema.

Esses degraus, medições relativas às entidades analógicas, precisam sofrer uma conversão a mais: precisam se transformar em bits. Essa transformação, que foi vislumbrada e modelada por Claude Shannon em sua Teoria da Informação², deve organizar esses degraus em linguagem binária (“zeros e uns”). Cada fragmento medido terá disponível um certo número pré-definido de zeros e uns para representá-lo. A possibilidade de se converter uma informação em zero ou um é o que se conhece por “*bit*”. Quanto maior for o número de *bits* disponíveis para essa representação, maior também será a resolução desse sistema. No caso de uma imagem, por exemplo, temos a resolução espacial (fragmentos da superfície da imagem) e uma resolução de cor e luminosidade (quantos *bits* cada fragmento da superfície tem para representar uma cor e seu brilho).

A determinação de quais sequências de zeros e uns são utilizadas para representar algo não obedece a qualquer relação causal ou indicial com o que representa: é puramente simbólica, por contiguidade (a informação não possuiria vínculo necessário com um conteúdo ou com um corpo/meio). Enfim, é arbitrária e, portanto, não há nada que poderíamos identificar em uma sequência binária que nos mostrasse se ela representa, por exemplo, uma cor ou um som. Uma mesma sequência numérica pode ser tratada como som ou como imagem pelo sistema. E aí reside algo sem precedentes na geração de entidades analógicas: podemos facilmente “ler” uma sequência binária que foi gerada a partir da conversão de um fragmento sonoro como sendo uma imagem, traduzindo essa sequência em variações de cores em uma superfície. O contrário também é válido: uma imagem pode ser lida como som.

Embora a conversão de som em imagem (e vice-versa) possa ser feita analogicamente, é com a digitalização que essa possibilidade se radicaliza: a recombinação das sequências binárias pode sintetizar entidades analógicas que não existiriam de outra maneira. A informação sem corpo poderia, portanto, assumir qualquer corpo ao ser trazida

² Teoria que trata da matematização dos processos de comunicação e transmissão de informação. Assim, estabelece critérios para medição de quantidade de informação que um canal pode transmitir, além de critérios para a sua modelação. Em: **Thomas, JA. Elements of Information Theory.** Wiley-Interscience, NewYork, 1991

ao universo analógico. Com isso, podemos mapear um domínio no outro, através de uma tradução que converte números “extraídos” de uma dimensão (sonora, visual, etc) em outra dimensão de natureza diferente. Esse mapeamento pode ser ainda mais elaborado: variações de intensidades podem ser mensuradas (a partir da digitalização) e posteriormente serem organizadas como parâmetros que alteram o comportamento de entidades de outra natureza. O mapeamento e a parametrização, processos esses que caracterizam a essência do Sistema Hidra(!), serão tratados mais adiante. Antes, porém, convém recuperar e desenvolver um conceito importante: a descorporificação da informação, isto é, a autonomia da informação quanto à matéria ou suporte. Essa autonomia propicia a própria existência do conceito de entrada e saída de dados (informações formatadas para cada caixa-preta) de uma caixa-preta. Essa autonomia propicia a própria existência do conceito de entrada e saída de dados em processos e cálculos digitais. Isso só é possível se tomarmos o sentido de informação nos moldes de Shannon. Nele, a informação não possui e não depende de um corpo. Ela apenas transita por corpos e qualquer ação desses corpos nessa transmissão deve ser prevista para se compor a informação. Se não for prevista, deve ser ao menos mensurável. Essa ação indesejada, mas mensurável, é o ruído. Ainda: a informação pode se utilizar de energia luminosa ou elétrica ou mecânica para se propagar, mas a informação em si não se altera. Uma mesma informação pode trafegar por todos esses meios e sua integridade deverá ser assegurada³.

Essa autonomização da informação é um conceito chave para entendermos os processos de digitalização e demais processos combinatórios digitais (que chamaremos de cálculos daqui em diante). Os *inputs* e *outputs* de um sistema digital (um computador, por exemplo) não são energia (luminosa, mecânica, elétrica) e sim informação. O *input* não é som, luz, sinal elétrico. O *input* é o som, a luz e o sinal elétrico transformados em informação. O *input* de um microfone (se entendido como uma caixa-preta) não é uma onda sonora (ou um conjunto delas). É a onda sonora tornada movimento mecânico que, por sua vez, é convertido em sinal elétrico. O que permanece nessas traduções é a informação. Assim, em concordância com a Teoria da Informação, a mesma informação ora é som, ora é movimento, ora é eletricidade.

No caso de uma máquina fotográfica, quando a imaginamos atuando normalmente, dizemos que ela está sendo sensibilizada pela luz. No entanto, o conceito de informação, como entendido pela Teoria da Informação, nos faz entender que a luz que penetra na câmera não é qualquer luz. É uma luz “organizada”: passa por uma lente num intervalo específico de

³ PAPER, Seymour. *Introduction to Embodiments of Mind* by Warren S. McCulloch. Cambridge, MA, M.I.T. Press, 1965. P.23.

tempo e viaja distâncias calculadas. Esse conjunto coordenado de ações sobre a luz a transformam em informação. É essa informação que reorganizará os materiais sensíveis à luz na superfície da película. Assim, a informação se transformou de luz organizada em composições químicas cristalizadas e organizadas. Essa informação, em outras palavras, não existia fora da máquina. Mas já existia em potência na máquina.

1.1 COMBINATÓRIA/POTÊNCIA/CÔMPUTO

Esses procedimentos, apresentados até agora em um contexto específico, dependem de condições que se estendem para qualquer caso envolvendo processos digitais e de programação. Assim, um sistema, para ser capaz de realizar cálculos, deve atender às seguintes premissas:

Deve ser matematicamente modelável:

Deve ter seu correspondente matemático, isto é, deve ser passível de ser traduzido em termos matemáticos. No entanto, como Kurt Gödel já demonstrou em seu “Teorema da Incompletude”⁴, nem tudo pode ser matematizável, no sentido de compreender uma verdade matemática. Gödel demonstra que não é possível uma prova da consistência e completude da matemática e que verdades matemáticas não são exclusivamente dedutíveis de um conjunto finito de axiomas iniciais, dependendo sempre da validação de um observador externo ao sistema⁵.

Deve ser algoritmicamente modelável:

Deve ser modelável em termos de uma máquina de Turing, com um conjunto finito de variáveis, estados e regras de funcionamento (instruções). E ao ser modelada como máquina de Turing, à ela se aplica o problema da “Parada de Turing”. Com o problema da

⁴ **GÖDEL, Kurt.** Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme, I. Berlin: Monatshefte für Mathematik und Physik, 1931.

⁵ **PENROSE, Roger.** *A Mente Nova do Rei*. Computadores, Mentas, e as Leias da Física. Rio de Janeiro: Ed. Campos, 1993.

Parada (*Halt Problem*), Turing define que, mesmo para sistemas deterministas como uma máquina de Turing, não existe a possibilidade de se prever quando um programa para (isto é, dá a resposta ao problema) antes de se efetuar todos os passos desse programa. Um sistema computável pode, assim, **ser determinístico e continuar sendo imprevisível.**

Deve ser logicamente modelável em termos da Lógica Booleana:

Em seu nível mais maquinal, mais próximo do hardware, as operações matemáticas se resumem à operações algébricas e lógicas. Essa camada, embora obliterada por interfaces mais amigáveis ao humano, são ubíquas e onipresentes em cada operação computacional.

Com a Lógica Booleana, que marcou a passagem da lógica aristotélica para uma lógica simbólica, proposições lógicas poderiam e deveriam ser expressas em termos de equações algébricas. Posteriormente, Claude Shannon mostrou que circuitos, como os encontrados numa máquina eletrônica, podiam ser expressos em termos de equações booleanas, isto é, estados fechados e abertos de interruptores de um circuito eletrônico poderiam corresponder a 0 ou 1, verdadeiro ou falso. Essa correspondência forneceu as bases para a construção de máquinas capazes de executar operações de lógica. Em nível teórico, Shannon apontou que a programação de um computador (a disposição de um conjunto de instruções codificadas para ser precisamente seguido) deveria ser tratada com um programa de lógica formal e não de aritmética (como era a tendência da abordagem aplicada nas máquinas computadoras/calculadoras na época). O que Shannon propôs, em outros termos, foi um mapeamento da lógica booleana em um domínio físico: lógica espelhada/traduzida em fluxo de energia organizada em um suporte material. Esse mapeamento ainda possuía um conceito fundamental: a informação poderia ser concebida como totalmente desvinculada de qualquer conteúdo específico, simplesmente como uma decisão única entre duas possibilidades igualmente plausíveis. A unidade básica que representaria essas possibilidades seria o “*bit*” (abreviação de “*binary digit*”). O *bit*, assim, seria a possibilidade em potencial de uma informação ser: “sim ou não”, “verdadeiro ou falso”, “ligado ou desligado”, “zero ou um”⁶.

⁶ Atualmente, com os computadores quânticos, existe a possibilidade de superação da bivalência dos estados booleanos 0 ou 1, verdadeiro ou falso, para estados intermediários. No entanto, esses estados intermediários são justaposições de estados clássicos da lógica booleana e seu impacto se dá apenas na velocidade dos passos a serem dados, pois estes podem ser calculados simultaneamente.

Deve ser tecnologicamente computável;

Como vimos em Turing e em Gödel, nem tudo pode ser modelado algorítmicamente, isto é, modelado de maneira a fazer parte de um programa no computador. O Teorema de Incompletude de Gödel e o Problema da Parada de Turing indicam que existe uma coerência intrínseca que limita e determina as possibilidades latentes em sistemas baseados na lógica simbólica. Esse universo de possibilidades se atualiza através de derivações a partir de construções (axiomas) iniciais. Essa coerência interna e autônoma, no entanto, sofre uma fricção contínua com o meio material onde ela acontece. É o atrito entre o lógico, entre o algoritmo, entre modelos científicos e a matéria, o tangível, o físico, o “corpo”, o meio.

Como vimos, na construção dos meios onde essa fricção acontece participaram nomes como Claude Shannon e o próprio Alan Turing. Além deles destaca-se a invenção da arquitetura dos computadores (memória, processador, etc.) por John von Neumann.

Deve existir um programador;

Insistimos aqui na presença de um programador, isto é, um sujeito (com suas limitações cognitivas, filtros ideológicos, etc.) que toma decisões na direção de atender as premissas relativas à coerência computacional, modelando computacionalmente um sistema capaz de automaticamente tomar decisões⁷ a partir das possibilidades finitas estabelecidas por um modelo pré-determinado.

A esse conjunto de premissas elencados acima chamamos de “coerência computacional”. Dela decorremos que:

- **O mundo do que é computável é um mundo determinístico** (de causas e efeitos, onde todas as possibilidades já estão contidas no “programa”); no entanto, esse mundo computável é imprevisível (segundo Turing).

- **Que em um cômputo novo, não há geração de informação nova.** Toda informação já está pré-estabelecida no programa. A informação e o cômputo são potenciais⁸, possibilidades pré-existent;

Entender o Sistema Hidra(!) a partir dessas premissas é, portanto, assumir que ele articula uma dimensão pré-determinável com um universo indeterminável (o uso humano da

⁷ De maneira mais precisa, seria mais adequado dizer que a tomada de decisão por parte do sistema é, portanto, nada mais do que a escolha dentre possibilidades pré-estabelecidas.

⁸ Potencial no sentido dado por Pierre Lévy em: LÉVY, Pierre. O que é virtual? São Paulo: Editora 34,1992.

interface). Ambos universos, para se comunicarem, precisam ser modelados em termos da coerência computacional.

O Sistema Hidra(!), assim, configura-se como mais uma camada combinatorial (existente em qualquer processo digital), no caso a mais externa delas, e que torna explícita essa propriedade relativa a geração de combinações.

2. SISTEMA HIDRA(!)

O Sistema Hidra(!) (o “!” aqui é uma notação matemática para “fatorial”, recurso utilizado em análise combinatória) promove a coordenação e a interação dos aspectos computacionais do ambiente e seus usuários. É importante ressaltar que assumimos o usuário/reprogramador como parte indissociável desse sistema interativo⁹. Para que essa interação se efetive, o Sistema Hidra(!) é dividido em três níveis: nível sensório, nível processador e nível atuador.

O nível sensório é formado por objetos técnicos que incorporam diversas formas de sensores de luz, de movimento, de temperatura, de distância, de umidade, de distância, etc.. Esses objetos técnicos, aliados a hardware específico (controladores como arduinos, raspberry pi, LittleBits e similares), convertem sinais analógicos em digitais. É o nível, portanto, onde se realiza a digitalização de sinais analógicos.

O nível processador é a parte que contém a programação, e que é reprogramável: conjunto de algoritmos que, esse nível processa essas informações digitais de diversas formas: reconhecendo padrões, tomando decisões (via operações booleanas simples e/ou complexas como Inteligência Artificial - I.A. - e algoritmos genéticos, por exemplo).

O nível atuador é formado por objetos técnicos que incorporam diversas formas de atuadores. Ele “atua” diretamente no ambiente através de interferências mecânicas (alterações controladas dos movimentos de elementos físicos/materiais desse espaço: abertura/fechamento de portas, janelas, brises, persianas e demais objetos-técnicos e interfaces móveis programáveis) e demais interferências físicas/tangíveis: síntese e manipulação sonora, controle luminoso, controle de temperatura e umidade, síntese e manipulação de imagens, etc. É um conjunto de atuadores (LEDs, buzzers, caixas de som, projetores, óculos de Realidade Virtual, lâmpadas, relays, servo-motores, motores de passo etc.).

⁹ Essa inclusão do usuário no sistema implica em novas questões e vemos na Cibernética de Segunda Ordem uma referência problematizante importante.

Esses níveis são programados utilizando-se a plataforma MAX7¹⁰. A programação desses níveis foi feita de modo que se propicie a recombinação e o mapeamento dinâmico entre sensores e atuadores, seja localmente ou remotamente (via internet). Ela possibilita, por exemplo, alterar o comportamento de um atuador (um LED, por ex.) a partir de um sensor de luz situado no mesmo local que o LED ou de um lugar muito distante, no caso, acionado por um controlador via internet¹¹.

A camada processamento do Sistema Hidra(!), como qualquer software programado na plataforma MAX7, possui dois estados principais: estado programação e estado operação. No primeiro, o usuário (ou designer de interface) pode reprogramar o software (que chamaremos de *patch* daqui em diante). Ele pode adicionar ou retirar funções e reorganizar o fluxo de dados. No modo operação o designer pode operar o *patch*, controlando-o, mudando variáveis, alimentando-o com dados que alteram o comportamento do *patch*. Temos, assim, dois níveis de interação possíveis: o da programação e da operação. Ambos podem ser complementares (quando se está programando um *patch*) ou, no caso do estado operação (quando se está controlando o *patch*), pode existir independentemente.

Começaremos ilustrando os componentes principais do software do Sistema Hidra(!), esclarecendo os passos iniciais para se configurar o sistema.

O software do Sistema Hidra(!) é composto por dois módulos principais: O primeiro, que recebe as informações provenientes dos sensores e as enviam pela internet aos outros terminais (sejam eles locais ou remotos). E o segundo, que recebe essas informações e possibilita a recombinação dinâmica delas, disponibilizando-as para que possam ser utilizadas como parâmetros para atuadores ou outros programas que possam receber esses parâmetros (Grasshopper, Unity 3D, Unreal4, por exemplo). Cada terminal da rede deve possuir os dois módulos, pois são complementares e interdependentes.

No primeiro módulo, configuramos a entrada de informações provenientes de sensores (do nível sensorial), indexando às variações numéricas termos que auxiliem na identificação da natureza da informação (se é sonora, luminosa, gasosa, distância, calor, etc.). Essa informação é empacotada junto com demais informações relevantes (qual o número do computador de onde parte essa informação, qual a porta que deverá ser ouvida

¹⁰ A utilização dessa plataforma se deu por esta utilizar uma linguagem estruturada em dataflow: os elementos da programação (as funções, variáveis, objetos, etc) são organizados graficamente e podem ser dinamicamente dispostos e relacionados entre si.

¹¹ Nesse mapeamento dinâmico inexistia a necessidade de carregar o arduino (um microcontrolador) com um novo programa a cada nova recombinação/ configuração/ mapeamento, o que, conseqüentemente, aumenta a velocidade das experimentações. Além disso, o sistema torna-se mais amigável à programação se comparado à interface convencional do arduino, baseada em linha de código e linguagem C/C++.

pelos outros terminais e se o sensor está operante ou não, fig. 1) e enviada para os computadores que fazem parte da rede local ou remota (fig. 2).

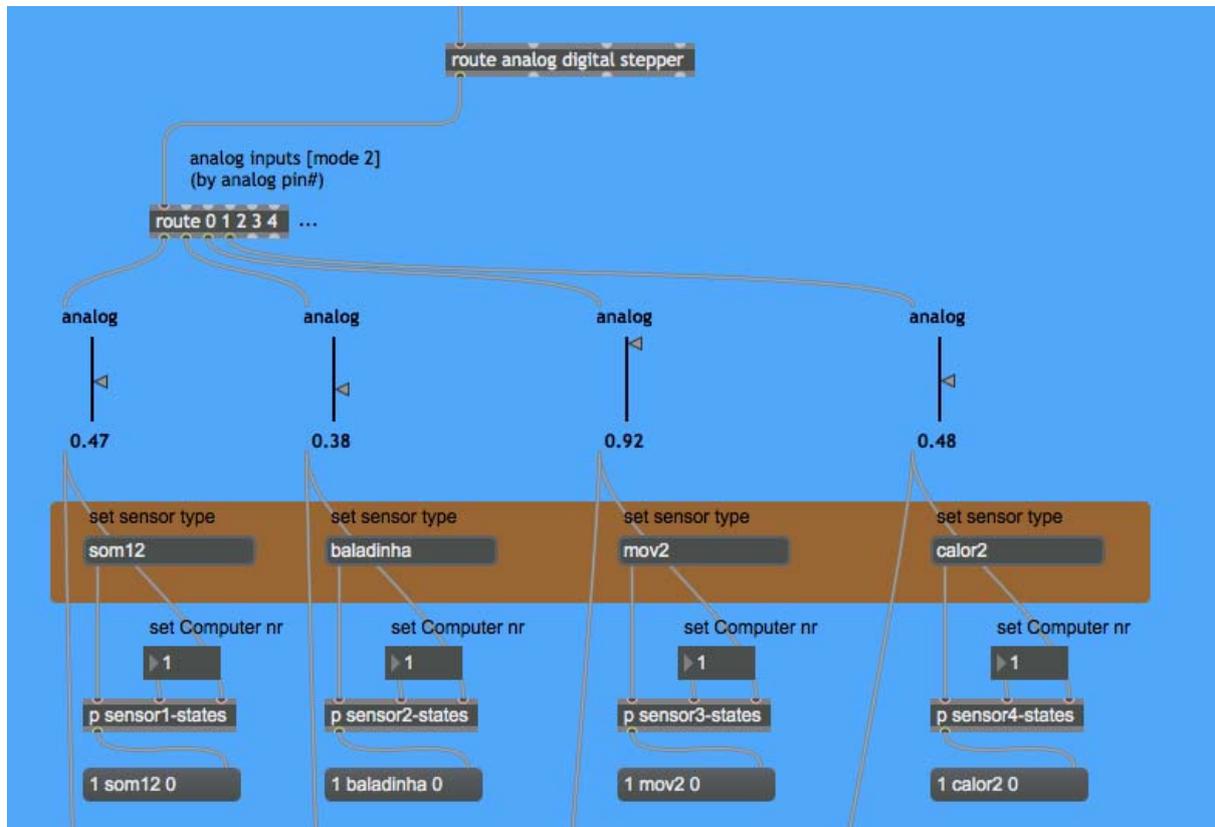


Figura 1 – Configuração das entradas de informação. Fonte: Autor, 2016.

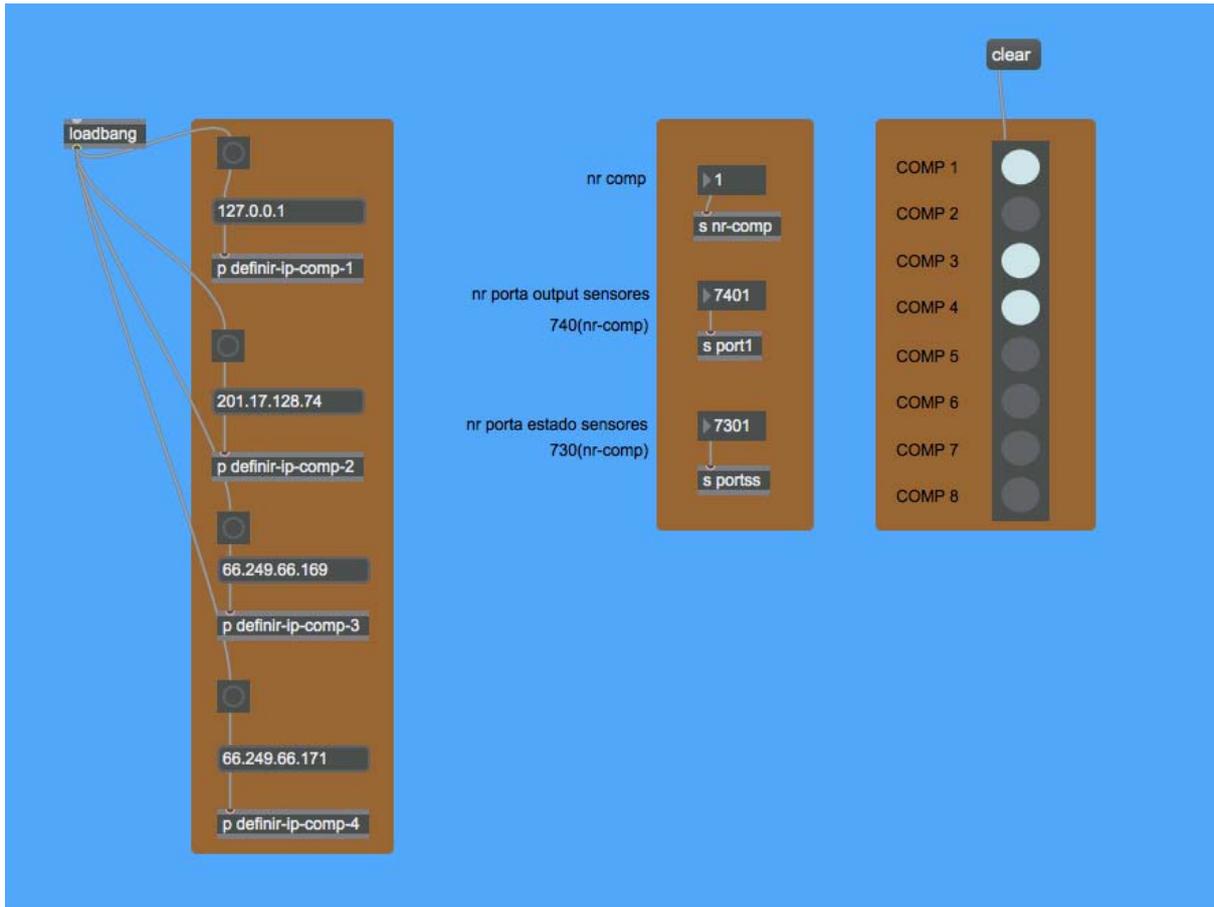


Figura 2 – Configuração dos endereços IPs dos computadores e definição de quais computadores participam da rede. Fonte: Autor, 2016.

No segundo módulo, o que recebe as informações da rede, é possível estabelecer de qual o computador iremos receber as informações. A informação que chega é “desempacotada” e seus elementos (informações relevantes sobre as origens de cada variação numérica) é tornada visível (fig. 3) e manipulável através de uma matriz que torna possível associar entradas de informação à diversas saídas (fig. 4). Esse módulo é a interface entre a camada combinatória do Sistema Hidra(!) e o nível atuador.

É nesse módulo onde a incorporação dos princípios da combinatória se tornam mais visíveis: é disponibilizada na interface um painel onde é possível recombinar, de maneira dinâmica e intuitiva, os diversos *inputs* vindos de sensores (e demais entradas) e os diversos *outputs* (atuadores, imagens de vídeo, modelos 3D parametrizáveis, etc).

2.1 EXPERIMENTOS:

Os experimentos relativos ao Hidra(!) se dividem em duas fases: a primeira, com ênfase técnica, são testadas diversas configurações dos elementos externos ao Hidra(!) (sensores, arduinos, atuadores, microfones, HTC Vive¹², definição dos protocolos de comunicação¹³ entre o Sistema Hidra(!) e demais softwares, no caso: Unity3D, Unreal4 e Grasshopper) e dos seus elementos internos (definição de protocolo de comunicação entre terminais com Sistema Hidra(!), definição da comunicação¹⁴ entre Arduinos e o Sistema Hidra(!), rotinas de calibragem automática de sensores, layout da interface em busca de uma maior amigabilidade).

Na segunda fase está prevista uma série de atividades ocupando três salas distantes entre si, onde serão exploradas dinâmicas de interação entre elas. Nessa fase a ênfase é na sensibilização dos alunos quanto aos aspectos combinatórios da interface e de como devemos “desenhar” os *inputs* enquanto informação a ser digitalizada para posterior cômputo, tornando tangível questões relativas ao conceito de “coerência computacional”.

No momento estamos finalizando a primeira fase com alguns aspectos da segunda fase já encaminhados.

Na primeira fase, por se tratar do design de aspectos técnicos do sistema, foi fundamental a consolidação dos conceitos aqui abordados por parte dos pesquisadores e alunos envolvidos: combinatória, digitalização, informação e, principalmente, o conceito de coerência computacional. Sem essa consolidação as estratégias relativas à organização do fluxo de informação, suas possibilidades combinatórias e otimização da distribuição da informação não teriam sido realizadas com clareza. E, principalmente, não teríamos alcançado um sistema com alto grau de abertura para sua re-programação (e

¹² Óculos de Realidade Virtual fabricado pela Vive: http://www.htcvive.com/us/develop_portal/

¹³ Optou-se pelo protocolo de comunicação UDP. Este protocolo já é bem disseminado como artifício para promover a comunicação entre softwares.

¹⁴ Utilizamos dois plug-ins para MAX7 que efetivam essa comunicação: Maxuino (em: <http://www.maxuino.org/>) e Arduivis (em: <https://github.com/cskonopka/arduivis>). Optou-se pelo Maxuino pela facilidade de utilização (maior amigabilidade da interface).

recombinação) por parte dos usuários. Podemos dizer, assim, que o Sistema Hidra(!) é a uma expressão, no formato de uma rede de aplicativos complementares e interconectados, dos conceitos aqui abordados.

Na segunda fase, iniciamos um trabalho de tensionamento desse sistema na direção de uma maior abertura para a interação. Nela buscaremos redesenhar a interface a partir de estudos de casos onde exploraremos diversas configurações de entrada de dados no sistema. Testaremos, portanto, a robustez e a escalabilidade do sistema, isto é, o quanto ele é ampliável sem perder sua coerência.

A interação com os ambientes interativos, assim, não se restringem em explorar configurações pré-estabelecidas e sim buscar reconfigurá-las buscando novas combinações e orquestrações entre *inputs* e *outputs*. Buscaremos explorar o caráter combinatório do Sistema Hidra(!) enquanto motor hibridizante através do redesenho do sistema a partir das demandas da interação, sendo que esse redesenho deve acontecer com a participação do interator/designer.

A seguir traremos um breve relato dos experimentos previstos na segunda fase.

2.2 EXPERIMENTOS PREVISTOS

Sensores e atuadores serão distribuídos em três ambientes situados em duas salas distantes umas das outras (sala do laboratório de Computação Radamés, Sala do Laboratório de Informática PCs e sala principal do Laboratório LAGEAR, sala). Tipos de sensores: luz, temperatura, distância, microfones, câmera, joysticks. Tipos de atuadores: caixas de som, LEDs, fitas LEDs, HTC Vive, projetores.

Na sala 1 (Radamés), projeta-se a imagem do ambiente virtual. Sensores de luz e calor distribuídos pela sala informam os vários estados da sala quanto a sua iluminação e temperatura.

Na sala 2 (LAGEAR), o HTC Vive também exibe o mesmo ambiente virtual (ambas imagens são referentes ao mesmo espaço/ambiente virtual. Cada terminal (projetor ou HTC Vive) tem a imagem atualizada a partir da perspectiva e controle da cada interator). Essa sala também conta com sensores de luz e microfones. Os microfones, associados a *patches* programados em MAX7, medem a variação de amplitude e frequência dos sons produzidos nas sala 2.

Na sala 3 (Lab PC), a ênfase é na produção de paisagens sonoras distribuídas em várias caixas de som (dez no total). Essas caixas formam um *grid* de pontos de saída de

áudio distribuídos por toda a sala. Em cada ponto também existe um sensor de distância que mede a proximidade do interator em relação à caixa de som.

O exercício proposto consiste em disponibilizar todos os *inputs* provenientes dos sensores para toda a rede. Assim, parâmetros de variação sonora (no caso da sala 3) e variações geométricas, de iluminação, cor e escala do avatar do interator em relação ao ambiente virtual (desde a escala do objeto até a escala de uma edificação de três andares) são remotamente gerados e localmente indexados aos *outputs* disponíveis. Assim, em um dado momento, a variação de luz da sala 1 pode parametrizar a variação de cor do ambiente virtual acessado na sala 2. Em um segundo momento, essa mesma variação de cor pode ser controlada pela variação de distância de um interator em relação a uma das caixas de som da sala 3.

Desta fase, já realizamos a comunicação entre os patches do Sistema Hidra e Um ambiente de Realidade Virtual programado no MAX 7, sendo prevista agora a utilização do software Unreal 4 para a geração de um ambiente mais elaborado e robusto.

Nesse experimento exploramos a variação dinâmica de *inputs* provenientes do ambiente (luz, som, movimento, etc.), indexando-os à parâmetros de uma superfície. Assim, transformamos a geometria (os vértices da geometria podiam ser manipuláveis), a escala, as cores e a iluminação do objeto virtual a partir desses parâmetros dinamicamente indexados via Sistema Hidra(!). Assim, quando o interator se movimentava pela sala, quando falava ou quando manipulava objetos luminosos, qualidades do objeto virtual se alteravam. O interator podia, portanto, observar, manipular ou visitar o objeto (a escala do objeto variava desde a escala de um objeto até a escala de uma edificação de três andares).

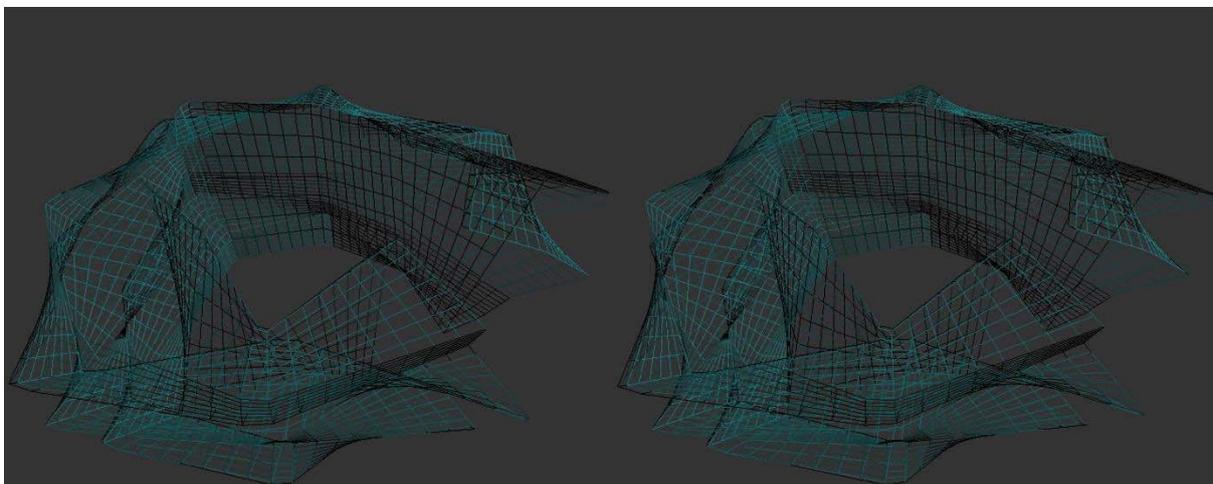


Figura 5 – Alteração da forma e das cores do objeto virtual a partir de *inputs* recombinaíveis via Sistema Hidra(!) – visão estéreo gerada para imersão no ambiente virtual. Fonte: Autor, 2016.

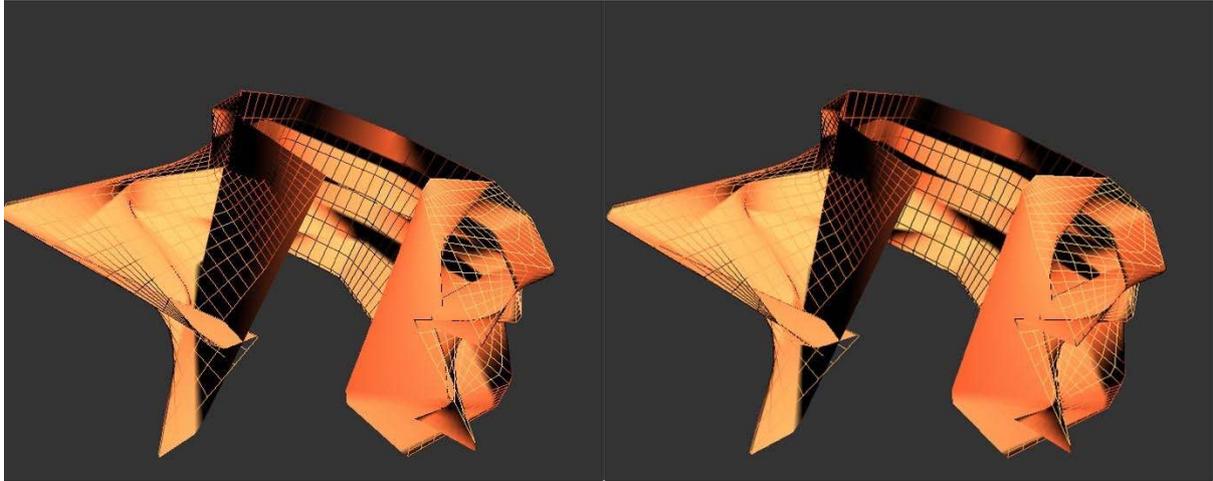


Figura 6 – Alteração da forma e das cores do objeto virtual a partir de *inputs* recombinaíveis via Sistema Hidra(!) – visão estéreo gerada para imersão no ambiente virtual. Fonte: Autor, 2016.

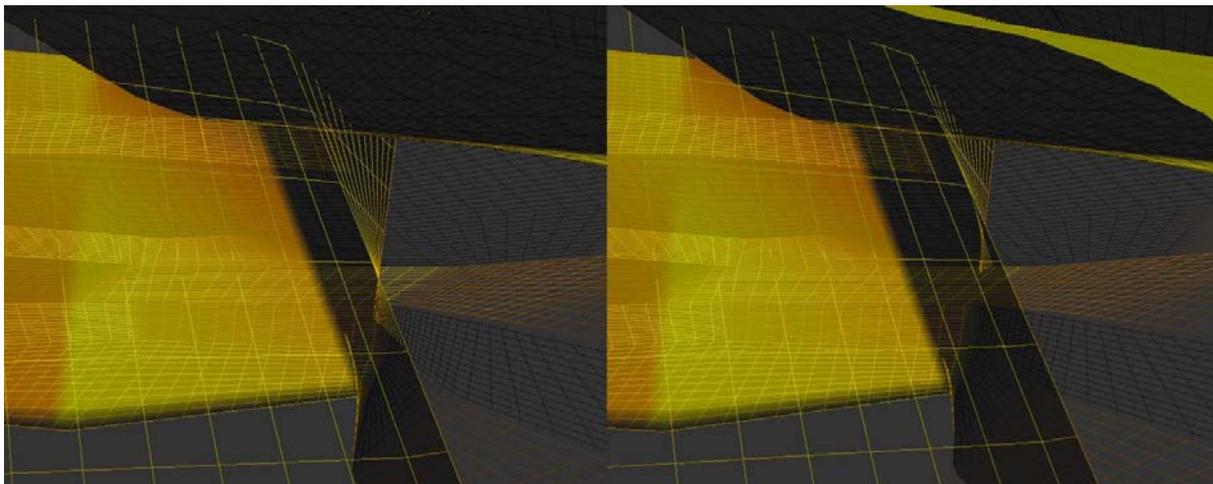


Figura 7 – Interator visitando internamente o objeto virtual – visão estéreo gerada para imersão no ambiente virtual. Fonte: Autor, 2016.

CONCLUSÃO

No desenho e sucessivos redesenhos do Sistema Hidra(!) o que está em jogo é a fricção entre o mundo determinístico da programação desse sistema (como vimos no

conceito de coerência computacional) e a possibilidade desse sistema (ao organizar e tornar tangível combinações possíveis entre informações captadas e processadas) de proporcionar apropriações novas, não previstas. Essa abertura para o novo só é possível na contínua tensão entre possibilidades latentes pré-determinadas e um uso/programação que explora intensamente a geração dessas combinações entre informações captadas. Nesse sentido, em essência, o Sistema Hidra(!) é um mero recurso determinístico que facilita a aparição de possibilidades latentes, pois fornece ao usuário a facilidade de transformar eventos captados de um ambiente em informação manipulável e recombinável. Assim, mais importante que a atual configuração desse sistema são os princípios que norteiam seu desenho, tornando-o, a um só tempo, manifestação tangível de um conjunto de conceitos (coerência computacional e combinatória) e ferramenta que possibilita a exploração desse conjunto ao explorar interações imprevisíveis. Dessa forma, o Sistema Hidra(!), em conjunto com o usuário/interator/programador, funciona como **catalizador de possibilidades latentes** que fazem sentido, que respondem a um problema ou ajudam a problematizá-lo.

Essa camada combinatória introduzida pelo Sistema Hidra(!) pode, portanto, potencializar a composição entre sistemas de computação física e internet das coisas ao disponibilizar a recombinação dinâmica entre seus elementos. Inputs provenientes de sensores podem ser endereçados a qualquer processo de computação vinculado ao Sistema Hidra(!). O resultado desses processos pode ser encaminhado a qualquer atuador dessa rede, proporcionando o controle de seu comportamento. Dessa forma, estados do sistema composto por terminais contendo módulos do Sistema Hidra(!) conectados à sensores, atuadores e demais softwares (Grasshoper, Unity 3D, etc) podem ser continuamente reorganizados.

A partir dos experimentos realizados e propostos, pretendemos consolidar estratégias e práticas que se adequem a um maior número de casos, incorporando essas soluções ao Sistema Hidra(!). Daremos ênfase aos experimentos remotos, conectando terminais situados em diversos Laboratórios de Informática com os quais estabeleceremos parcerias. Nesse sentido, organizaremos *kits* de equipamentos e softwares e promoveremos workshops para a difusão dessas práticas e tecnologias.

BIBLIOGRAFIA

ABREU, S. C. Interfaces em Arquitetura: Permeabilidades entre o Humano e o Digital. Ed. USP. São Carlos, 2011.

LATOUR, Bruno. Jamais Fomos Modernos. 3ª reimpressão, São Paulo: Editora 34, 2005.

LÉVY, Pierre. O que é virtual? São Paulo: Editora 34, 1992.

McLUHAN, Marshall. Understanding Media: The Extensions of Man. MIT Press, 1994.

PAPERT, Seymour. Introduction to Embodiments of Mind by Warren S. McCulloch. Cambridge, MA, M.I.T. Press, 1965.

PENROSE, Roger. A Mente Nova do Rei. Computadores, Mentas, e as [Leis](#) da Física. Rio de Janeiro: Ed. Campos, 1993.

SANTOS, Laymert Garcia dos. Politizar novas tecnologias. O impacto sócio-técnico da informação digital e genética. Estética e Política. São Paulo: Editora 34, 2003.

SHANNON, C. E. A. Symbolic Analysis of Relay and Switching Circuits. Cambridge: MIT Press, 1940.

SIMONDON, G. L'individu et sa genese plzysicobiologiqlle .Grenoble: Editions Jerome Millon, 1995.

_____. **On the Mode of Existence of Technical Objects,** Trans. by N. Mallahphy, London: University of Western Ontario, 1991.

Thomas, JA. Elements of Information Theory. New York: Wiley-Interscience, 1991